Portable Intrusion Detection System

Harshitha Prasad Rao, Anirudh M.R, Diksha U.S, Bhuvana Suganthi D

Abstract- Every network connection made is exposed to a security threat. Due to this, crucial, confidential, important information can be lost or even exploited by an intruder. Hence an intrusion detection system (IDS) is needed to detect and notify the network administrators if such an event occurs. Though such systems are present in organizations, it is confined to a device or a rack. In this proposed system, an open source network intrusion detection system called SNORT is installed on to a Raspberry Pi. This is then implemented at a switch level, where it is connected to a configured SPAN port of a switch and is used to monitor the traffic through the switch. This system provides logs and information of the traffic, as well as notifies the network administrator through a message in real time. The intrusion detection system's portability, ability to notify the administrator, and the display of packet information is what makes this system, more advanced and convenient for any organization's security.

Keywords: Intrusion, NIDS, SNORT, SPAN, Security

I. INTRODUCTION

Organizations have a computer network through which transfer of data takes place. This information can sometimes be crucial and confidential. Due to this economic value, organizations must prevent data exploitation. It was observed that the global average cost of a data breach is $3.92 million. The average cost of lost business for organizations in 2019 was $1.42 million, which represents 36% of the total average cost of $3.92 million [1]. This being the scenario, the need of network security arises. The current situations to tackle this problem are software applications which are installed on each individual system and the monitoring takes place. But physically, monitoring the network all the time is not practical and the need for automation arises. To prevent such attacks, it must be detected first. The proposed system, which is a hardware device, can be used to detect such intrusions and alert the administrator. The device/system proposed is advantageous over the existing software in the following ways:

- It is portable and can be used at any L2 device.
- It detects attacks and gives detailed information about the malicious traffic.
- The Graphical User Interface makes it easier to analyze the vulnerability.
- Saves time and resources.

II. LITERATURE SURVEY

The usage of networks has increased over the years. Security measures must be implemented to protect these networks. One such method is an intrusion detection system. As it is placed in line with the network it is called a network intrusion system. The system sniffs and monitors the network for violations corresponding to confidentiality, integrity and availability [2]. If any violations found it considers it as an intrusion. Different methods are used to detect the intrusions which are signature, anomaly and stateful protocol [3]. A signature-based IDS compares patterns against a database of predefined strings, it's incapability to detect unknown attacks is balanced with an anomaly detection IDS which detects a deviation in the behavior as an attack. While comparing the different methods, a combination of signature based and anomaly-based detection methods is preferred. There has been work related to this field by the following authors Hung-Jen et al., [3] Signature based detection compares patterns against the captured events for recognizing an attack; it cannot detect a new type of attack, that is one which is not present in the database. Anomaly detects a deviation in the network behavior as an attack. The main disadvantage is the accuracy of the alerts. A software tool called SNORT which is a combination of a signature based and anomaly-based IDS having a rule-based approach is used. Garcìa-Teodoro et al., [4] Study was made to understand in depth on how anomaly-based intrusion detection systems work, which is the methodology implemented by the software we have used, Snort. Anomaly based detection identifies any deviation in behavior for example, failed login attempts at random intervals of the day. The limitations of this method include the difficulty to trigger an alert on time and also the false positives that is it sometimes detects a normal packet as an intrusion or can even fail to detect an intrusion. Martin et al., [5] Snort is nimble and can work very quickly. It has real time alerting capability, which means that as soon as the attack is detected it, alerts the network administrator. Once it has been configured, it doesn't need to be checked every now and then. It's a packet sniffer and a logger also it can work in-line and bridge mode.

A. Problem statement

Every network connection made is exposed to a security threat. Due this, crucial, confidential, important information can be exposed or even lost to a third-party intruder. Hence, in order to address, this paper aims at detecting if an intrusion is taking place in the network and also to alert the network administrator so that they can take up necessary alleviating actions. There are many appliances in the market that detect and alert,

Revised Manuscript Received on June 30, 2020.
* Correspondence Author

Harshitha Prasad Rao*, Student, Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India. Email:harshitha_rao@hotmail.com

Anirudh M.R, Student, Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India. Email:anirudhmr1998@gmail.com

Diksha U.S, Student, Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India. Email:dikshaus22@gmail.com

Dr. Bhuvana Suganthi D, Associate Professor, Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India. Email:bhuvanasuganthi@gmail.com

Retrieval Number: E1218069520/2020©BEIESP
DOI: 10.35940/ijeat.E1218.069520
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication
© Copyright: All rights reserved.

but two common problems with them are that they are bulky, have to be installed in a fixed place and are expensive. The proposed method is to eliminate the above limitations using SNORT method.

B. SNORT

Snort is an open source network intrusion detection system (NIDS) created by Martin Roesch [5]. Snort is a packet sniffer that monitors network traffic in real time, performing deep packet inspection that is, analyzing each packet closely to detect a dangerous payload or suspicious anomalies. Snort's open source network-based intrusion detection/prevention system (IDS/IPS) has the ability to perform real-time traffic analysis and packet logging on Internet Protocol (IP) networks. Snort analyses the protocol, performs content searching and matching.

III. METHODOLOGY

A Raspberry Pi 4 forms the core of the portable system. It is loaded with an Ubuntu operating system and is connected to a switch through an Ethernet interface. A switch helps in directing the traffic from a source to the intended destination port. For the IDS device to acquire and analyse the data packets, all the network traffic must reach the port to which the IDS is interfaced to. To facilitate this, the IDS port is configured to work in the Switched Port Analyser (SPAN) mode. This directs all the network traffic to the port allowing the IDS to sniff the traffic. This can be configured through the Command Line Interface (CLI) of the switch. The promiscuous mode of the Ethernet interface is enabled so that the IDS can process the data without being the destination port. Hence the SPAN mode of the port and the promiscuous mode of the Ethernet interface allow the IDS to monitor the traffic. Wireshark and SNORT tools are loaded in the Raspberry Pi. Wireshark is used to analyse the data packets in the network which gives detailed information about the traffic. SNORT is to be setup first and it is to be operated in the intrusion detection mode. SNORT comes with a power package of a number of signatures and rules which are predefined. New rules can also be written to detect for a specific type of traffic. With the setup in place, a copy of the traffic is sent to the SPAN port. The IDS device analyses the packets and checks for malicious content and creates log files. The data in the log file is to be displayed on the GUI for better understanding and analysis. With python programming this data is sent to a monitoring system where it can be viewed and it can also alert the concerned person about the attack through an SMS. The flow of work process is shown in figure 1.

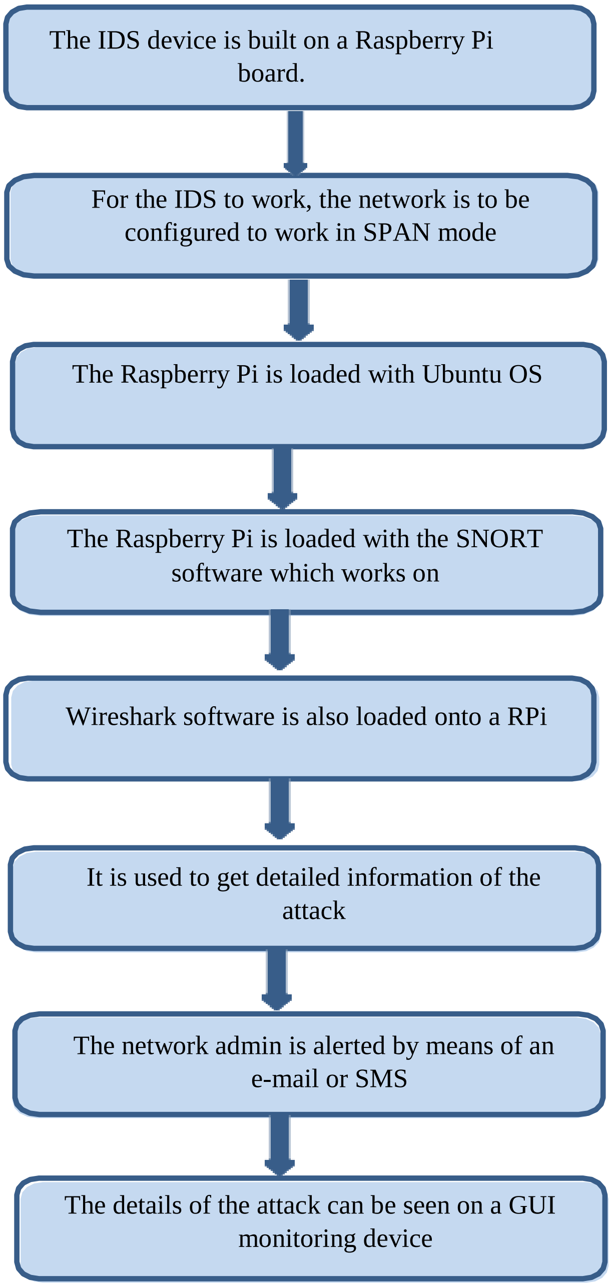

The IDS device is built on a Raspberry Pi board.
For the IDS to work, the network is to be configured to work in SPAN mode
The Raspberry Pi is loaded with Ubuntu OS
The Raspberry Pi is loaded with the SNORT software which works on
Wireshark software is also loaded onto a RPi
It is used to get detailed information of the attack
The network admin is alerted by means of an e-mail or SMS
The details of the attack can be seen on a GUI monitoring device

Fig.1. Flowchart of the IDS working

Network based Intrusion Detection System is connected to a switch. This helps in improving the visibility of the malicious activity in the network. The block diagram of network intrusion detection system is shown in figure 2.

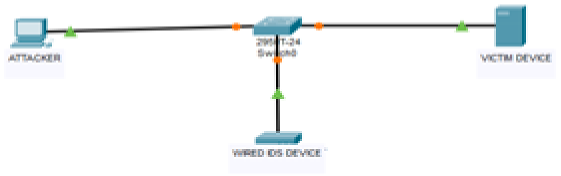
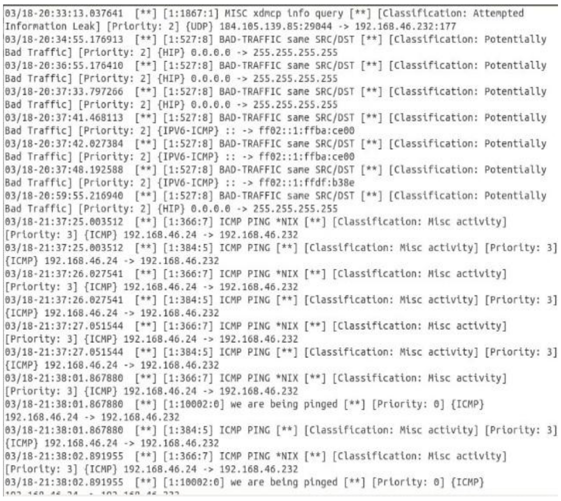


Fig.2. NIDS Block Diagram

IV. RESULT

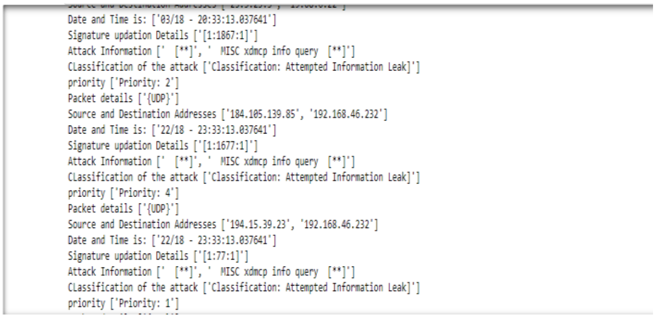
SNORT is used to monitor the network. If an attack occurs, it gets logged in an alert file. Once the alert file is updated, a SMS is triggered and an email is sent to the network administrators. This helps them to take the necessary steps in combating the attack. The alert file of the SNORT is in the form of logs. In order to display the details of an attack in a more understandable format, a graphical user interface is created which parses each log of the alert file, extracts different parameters of the file, like date, time, signature details, attack information, packet details, source and destination addresses. This GUI gets updated in real time with an attack. It displays an alert saying “An attack has occurred” and has a stop timer. The alert file, email, SMS, and GUI are shown below. The alert file generated after running SNORT in IDS mode is shown in figure 3.



```
03/18-20:33:13.037641  [**] [1:1867:1] MISC xdmcp info query [**] [Classification: Attempted
Information Leak] [Priority: 2] {UDP} 184.105.139.85:29044 -> 192.168.46.232:177
03/18-20:34:55.176913  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially
Bad Traffic] [Priority: 2] {HIP} 0.0.0.0 -> 255.255.255.255
03/18-20:36:55.176410  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially
Bad Traffic] [Priority: 2] {HIP} 0.0.0.0 -> 255.255.255.255
03/18-20:37:33.797266  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially
Bad Traffic] [Priority: 2] {HIP} 0.0.0.0 -> 255.255.255.255
03/18-20:37:41.468113  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially
Bad Traffic] [Priority: 2] {IPV6-ICMP} :: -> ff02::1:ffba:ce00
03/18-20:37:42.027384  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially
Bad Traffic] [Priority: 2] {IPV6-ICMP} :: -> ff02::1:ffba:ce00
03/18-20:37:48.192588  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially
Bad Traffic] [Priority: 2] {IPV6-ICMP} :: -> ff02::1:ffdf:b38e
03/18-20:59:55.216940  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially
Bad Traffic] [Priority: 2] {HIP} 0.0.0.0 -> 255.255.255.255
03/18-21:37:25.003512  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity]
[Priority: 3] {ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:37:25.003512  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3]
{ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:37:26.027541  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity]
[Priority: 3] {ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:37:26.027541  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3]
{ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:37:27.051544  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity]
[Priority: 3] {ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:37:27.051544  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3]
{ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:38:01.867880  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity]
[Priority: 3] {ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:38:01.867880  [**] [1:10002:0] we are being pinged [**] [Priority: 0] {ICMP}
192.168.46.24 -> 192.168.46.232
03/18-21:38:01.867880  [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3]
{ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:38:02.891955  [**] [1:366:7] ICMP PING *NIX [**] [Classification: Misc activity]
[Priority: 3] {ICMP} 192.168.46.24 -> 192.168.46.232
03/18-21:38:02.891955  [**] [1:10002:0] we are being pinged [**] [Priority: 0] {ICMP}
```

Fig.3. SNORT Alert File

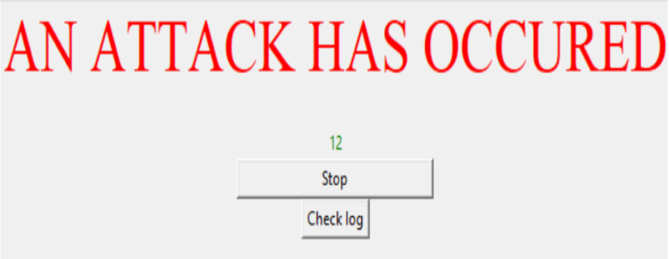
The contents of the SNORT alert file are difficult to understand and the various parameters are classified as shown in figure 4 which separates the date and time, the signature ID, the attack type, priority of the attack and also the source and destination address.



```
Date and Time is: ['03/18 - 20:33:13.037641']
Signature updation Details ['[1:1867:1]']
Attack Information ['  [**]', '  MISC xdmcp info query  [**]']
CLassification of the attack ['Classification: Attempted Information Leak]']
priority ['Priority: 2']
Packet details ['{UDP}']
Source and Destination Addresses ['184.105.139.85', '192.168.46.232']
Date and Time is: ['22/18 - 23:33:13.037641']
Signature updation Details ['[1:1677:1]']
Attack Information ['  [**]', '  MISC xdmcp info query  [**]']
CLassification of the attack ['Classification: Attempted Information Leak]']
priority ['Priority: 4']
Packet details ['{UDP}']
Source and Destination Addresses ['194.15.39.23', '192.168.46.232']
Date and Time is: ['22/18 - 23:33:13.037641']
Signature updation Details ['[1:77:1]']
Attack Information ['  [**]', '  MISC xdmcp info query  [**]']
CLassification of the attack ['Classification: Attempted Information Leak]']
priority ['Priority: 1']
```

Fig.4. Classifying the parameters of the alert file

The GUI window displaying a message with a stop timer is shown in figure 5.

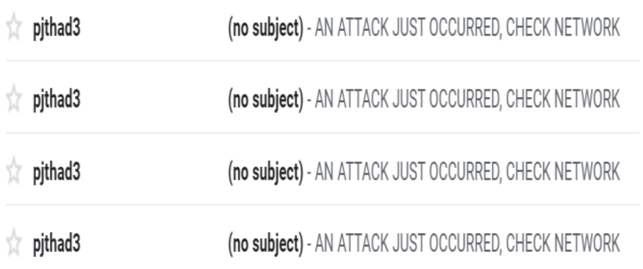


AN ATTACK HAS OCCURED

12

Fig.5. An alert with Stop Timer

After an attack has occurred, an alert Email is sent to the network administrator. A snapshot of the Email received is shown in figure 6.



☆ pjthad3 (no subject) - AN ATTACK JUST OCCURRED, CHECK NETWORK

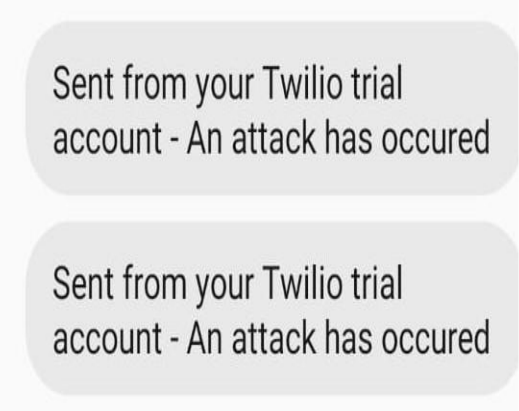
☆ pjthad3 (no subject) - AN ATTACK JUST OCCURRED, CHECK NETWORK

☆ pjthad3 (no subject) - AN ATTACK JUST OCCURRED, CHECK NETWORK

☆ pjthad3 (no subject) - AN ATTACK JUST OCCURRED, CHECK NETWORK

Fig.6. Email sent to the Administrator

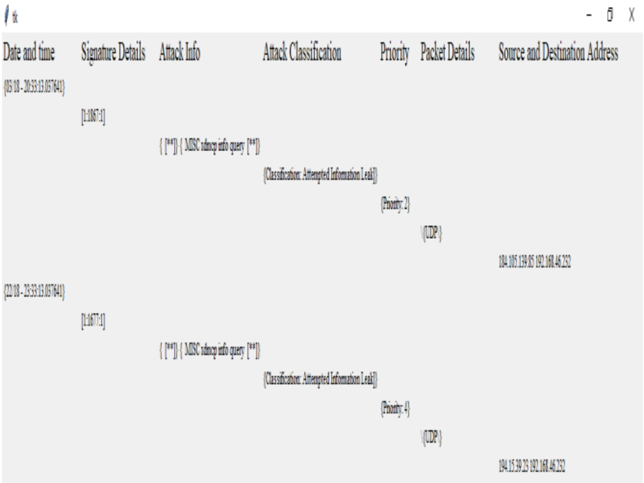
A SMS is sent to the network administrator. The message received is shown in figure 7.



Sent from your Twilio trial account - An attack has occured

Sent from your Twilio trial account - An attack has occured

Fig.7. SMS sent to the Administrator



Date and time	Signature Details	Attack Info	Attack Classification	Priority	Packet Details	Source and Destination Address
{05/18 - 20:33:13.037641}						
	[1:1867:1]					
		{ [**] } { MISC xdmcp info query [**] }				
			{Classification: Attempted Information Leak]}			
				{Priority: 2}		
					\{UDP }	
						184.105.139.85 192.168.46.232
{22/18 - 23:33:13.037641}						
	[1:1677:1]					
		{ [**] } { MISC xdmcp info query [**] }				
			{Classification: Attempted Information Leak]}			
				{Priority: 4}		
					\{UDP }	
						194.15.39.23 192.168.46.232

Fig.8. GUI of the attack details

For a better understanding of the attacks, a GUI is created as shown in figure 8. This table is updated with the logs of the alert file as shown in figure 4.

V. CONCLUSION

Organization networks have vulnerabilities which are exploited by hackers. Crucial and sensitive information can be lost or altered in the process. Hence there is a need of a system to detect these attacks and notify the network administrators. SNORT is installed onto a Raspberry Pi. It is connected to a SPAN port of the switch which allows it to read the traffic. Once an intrusion occurs it is logged and a SMS and email are sent to the considered authority. Information regarding the intrusion is displayed on a GUI and packet analysis is done by Wireshark. This system is portable light weighted and is user friendly as it is preconfigured for monitoring purposes. This makes it more helpful than the present-day methods used.

FUTURE WORK

The accuracy of the system can be improved with artificial intelligence and honeypots [6]. A more secure and accurate method can be devised like

- Accuracy: False alarms are generated through different approaches of the IDS. Hence the accuracy must be improved.
- It works only for wired networks. Fails to accommodate the bandwidth associated with wireless network
- Parallel Computing can be introduced, which will bring down the time and overhead of the intrusion detection system. The system can be improvised through machine learning and artificial intelligence. This would also be able to take care of zero-day attacks and advanced persistent threats.

REFERENCES

1. IBM Security's Data Breach Report 2019.
2. Sailesh Kumar, "*Survey of Current Network Intrusion Detection Techniques"*, Washington University,2009.
3. Hung-Jen Liao, Chun-Hun Richard Lin, Ying-Chih Lin, Kuang-Yuan Tung, "*Intrusion detection system: A comprehensive review*", Elsevier,2008.
4. Garcìa-Teodoro, J. Dìaz-Verdejo, G.Macià-Fernàndez,E.Vàzquez, "*Anomaly-based network intrusion detection: Techniques, systems and challenges*", Elsevier,2008.
5. Martin Roesch, "*Snort–Lightweight Intrusion Detection for*
6. *Networks"*, Stanford Telecommunications Inc,1999
7. E. Bharathi, M. Keerthana, G. Ramsundar, "*Intrusion Detection Using Raspberry Pi Honeypot*", IRJET,2019

AUTHORS PROFILE

Harshitha Prasad Rao, Under graduate student, Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India.
Email: harshitha_rao@hotmail.com

Anirudh M.R, Under graduate student, Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India.
Email: anirudhmr1998@gmail.com

Diksha U.S, Under graduate student, Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India.
Email: dikshaus22@gmail.com

Dr. Bhuvana Suganthi D, Associate Professor**,** Department of Electronics and Communication Engineering, B.N.M Institute of Technology, Bangalore, India.
Email: bhuvanasuganthi@gmail.com